



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/724,921	12/01/2003	Kohsuke Kawaguchi	5681-70000	1840

35690 7590 07/02/2007
MEYERTONS, HOOD, KIVLIN, KOWERT & GOETZEL, P.C.
P.O. BOX 398
AUSTIN, TX 78767-0398

EXAMINER

KHATRI, ANIL

ART UNIT	PAPER NUMBER
----------	--------------

2191

MAIL DATE	DELIVERY MODE
-----------	---------------

07/02/2007

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/724,921	Applicant(s) KAWAGUCHI ET AL.	
	Examiner Anil Khatri	Art Unit 2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 01 December 2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-61 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-61 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 01 December 2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date <u>7/7/06</u> | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

Specification

The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.

The following title is suggested: "*Proxying Mechanism and Isomorphic Interfaces in Subsystem Within Virtual Machine Environment*".

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claims 1-61 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1- 61 are rejected under 35 USC 101 because they disclose a claimed invention that is an abstract idea as defined in the case *In re Warmerdam*, 33, F 3d 1354, 31 USPQ 2d 1754 (Fed. Cir. 1994).

Art Unit: 2191

Analysis: Claims 1-61 disclosed by the applicant as being a “system and method...”. Since the claims are each a series of steps to be performed on a computer the processes must be analyzed to determine whether they are statutory under 35 USC 101.

Examiner interprets that the claims 1-61 are non-statutory because they do not disclose that how a system will be able to carry out and execute its intend result without incorporating a processor, memory and medium. Further, examiner interprets that claims 1-61 are non-statutory because claim recites computer program for configuring are, per se i.e. the description or expressions of the program are not physical things nor are they statutory process as they do not act being performed. Computer programs do not define any structural and functional interrelationship between the computer program and other claimed aspect of the invention which permits the computer program’s functionality could be realized. Therefore, computer program is merely a set of instructions capable of being executed by a computer, the computer program itself is not a process. Thus claims 1-61 are non-statutory and rejected under 35 USC 101.

Analysis: Claims 45-61 disclosed by the applicant as being a “a computer accessible medium...”. Since the claims are each a series of steps to be performed on a computer the processes must be analyzed to determine whether they are statutory under 35 USC 101.

Examiner interprets that claims 45-61 are not limited to tangible embodiments in view of applicant’s disclosure, specification pages 30-31, lines 29-30 and 1-6 the medium is not limited to tangible embodiments, instead being defined as including both tangible embodiments (e.g., [computer readable medium]) and intangible embodiments (e.g., [transmission media, radio frequency (RF), infrared (IR), a carrier wave, telephone line, a signal, etc.]). As such, the claim is not limited to statutory subject matter and is therefore non-statutory. To overcome this type of

Art Unit: 2191

101 rejection the claims need to be amended to include only the physical computer media and not a transmission media or other intangible or non-functional media. For the specification at the bottom, carrier medium and transmission media would be not statutory but storage media would be statutory.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1-61 are rejected under 35 U.S.C. 102(e) as being anticipated by *Krapf et al*

USPN 6,901,588.

Regarding claims 1-6, 13-15, 18, 20-25, 27-32, 38, 39, 41-43, 45-49, 53, 55, 56 and 58-60

Krapf et al teaches

a plurality of subsystems configured to execute within a virtual machine on the system, wherein two or more of the subsystems each provide a version of an isomorphic interface to functions of the subsystem (column 39, lines 6-16, A C++ proxy class may wrap a Java Listener interface for the Abstract Window Toolkit (AWT) or Swing subsystems. In Java, an Adapter typically provides a do-nothing default implementation of a Java Listener interface. Thus, Adapters are inherently useless unless used as a base class for a concrete Listener implementation. Using an Adapter as a base class allows the Java developer to selectively override a single method of the Java Listener interface rather than implement all methods that the Java Listener interface

Art Unit: 2191

declares. Consequently, a C++ developer may wish to override one or more Adapter methods in C++ ; and

a proxy mechanism configured to generate a proxy to one of the two or more subsystems that provides a correct version of the isomorphic interface for one of the plurality of subsystems at runtime of the one of the plurality of subsystems, wherein the proxy is configured to (column 27, lines 46-48, Assignment statements 258-266 (of FIG. 11) achieve correct results through the same proxy mechanisms that cause assignment statement 256 to operate as the developer intended);

receive a call to the isomorphic interface from the one of the plurality of subsystems;
convert the call in accordance with the version of the isomorphic interface provided by the one of the two or more subsystems (column 42, lines 13-16 If, however, a concrete C++ class overrides the C++ class' callback into Java, the concrete C++ class's version of the callback method gets called as described above in connection to FIG. 16, thereby overriding the Java class's implementation; and

forward the converted call to the one of the two or more subsystems for execution (column 43, lines 49-61, In a first strategy, JNI exceptions and errors may be cleared in a JNI layer, and execution of a shared C++ and Java application may effectively ignore the exception or error. Using this strategy, if a high-level C++ concept translates into a sequence of JNI API invocations, the failure of any one JNI call in the sequence may cause the remainder of the sequence to not be executed. C++ proxy classes may not be provided for Java exceptions because exceptions are not used. Thus, this strategy involves a high level of risk because

Art Unit: 2191

failures occur silently. Consequently, because most Java methods throw concrete exceptions to signal certain conditions to the calling code, this error-handling strategy may limit semantic usability of C++ proxy components that merely clear JNI exceptions and errors in the JNI level).

Regarding claims 7

Krapf et al teaches

proxy mechanism is further configured to provide an API to the subsystems, wherein the API is configured for use by the subsystems to specify isomorphic interfaces to be proxied to the proxy mechanism (column 8, lines 57-67, If a component is shared between two domains, a C++ proxy component representing a first concept has a semantic usability closely corresponding to the semantic usability of a Java component representing the first concept. In one implementation, the Java Native Interface (JNI), a Java Application Programming Interface (API), may be used to code the proxy layer of the C++ proxy component. In another implementation, other interfaces such as, for example, Microsoft's Raw Native Interface (RNI) or Netscape's Java Runtime Interface (JRI), may be used to code the proxy layer of the C++ proxy component. In either implementation, the interface being used is supported by the targeted Java Virtual Machine (JVM).

Regarding claims 8, 17, 26, 33, 40, 50 and 57

Krapf et al teaches

proxy is further configured to convert the call in accordance with the version of the isomorphic interface provided by the one of the two or more subsystems using Java Reflection (column 15, lines 12-24, Java permits Java classes to be defined final (as Java also permits for methods, which will be discussed in more detail below). The Java semantics dictate that a Java class defined to be final is not allowed to be a superclass for other classes. In other words, the Java class is not inheritable. C++ does not have a corresponding concept to this final definition that restricts inheritability of C++ classes. This absence of the final concept in C++ does not have adverse effects on the transformation per se. For example, a C++ proxy class may include a comment reflecting the intended inheritability of the C++ proxy class. Although this comment can not be enforced by the C++ compiler, it may inform a C++ developer of the intended non-inheritability).

Regarding claims 9, 19, 25, 34, 44 and 51

Krapf et al teaches

the virtual machine is a Java Virtual Machine (JVM) (column 9, lines 57-67, Three particular aspects of JNI that impact generating C++ proxy components are: JNI thread management; JNI object-reference types; and JNI context-dependency. JNI thread management imposes a requirement that a thread originally created in a C or C++ program must explicitly be attached to a JVM before other JNI calls can be made on this processing thread. Most JNI

Art Unit: 2191

function invocations take a JNIEnv pointer as an argument. The JNIEnv pointer is specific to a thread, i.e., attaching the thread to the JVM returns a JNIEnv pointer that must not be used from other threads).

Regarding claims 10, 35 and 52

Krapf et al teaches

the one of the plurality of subsystems is an application, and wherein the two or more subsystems are versions of a runtime library (column 44, lines 38-49, Typical implementations of Java Virtual Machines reside in a shared library. In order to load a Java Virtual Machine, the shared library containing the JVM is identified. In an optional aspect of this embodiment, the shared library containing the JVM may be identified through a configuration setting. After the shared library is loaded, the JVM has to be started and initialized. JVM initialization arguments may include settings like the classpath, available RAM and other information that may vary between JVMs. In another optional aspect of this embodiment, initialization arguments may be specified through configuration settings).

Regarding claims 11, 36 and 61

Krapf et al teaches

the one of the plurality of subsystems and the two or more subsystems are applications (see abstract).

Art Unit: 2191

Regarding claims 12, 37 and 54

Krapf et al teaches

wherein the plurality of subsystems are mobile (column 34, lines 55-67, any of the above mutability mappings may be customized on a method-by-method basis. In other words, a developer may choose to define a mutability attribute for a C++ component different than those described above. If a default mapping of a mutability signature for a C++ proxy method is customized, a mutability signature of all C++ proxy methods that override such C++ proxy method should be customized accordingly to maintain correct overriding semantics. Such customization may be applied to any overriding methods and any C++ subclass method of such C++ proxy method. Further, the customization may be applied to any superclass' methods overridden by such C++ proxy method, but only if the overridden method is a virtual method, which ensures that C++ virtual semantics are not violated. If, in contrast, the overridden super class method is non-virtual, the customization may not be propagated, and the overridden super class method may be effectively hidden by such a C++ proxy method, which differs from the overridden super class method only in consents.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Anil Khatri whose telephone number is 571-272-3725. The examiner can normally be reached on M-F 8:30-5:00 PM.

Art Unit: 2191

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Zhen can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.


ANIL KHATRI
PRIMARY EXAMINER